EXPERIMENT - 4

Object Oriented Programming Lab

Aim

Using concept of function overloading, write function for calculating area of triangle, circle and rectangle.

Syeda Reeha Quasar
14114802719
4C7

EXPERIMENT – 4

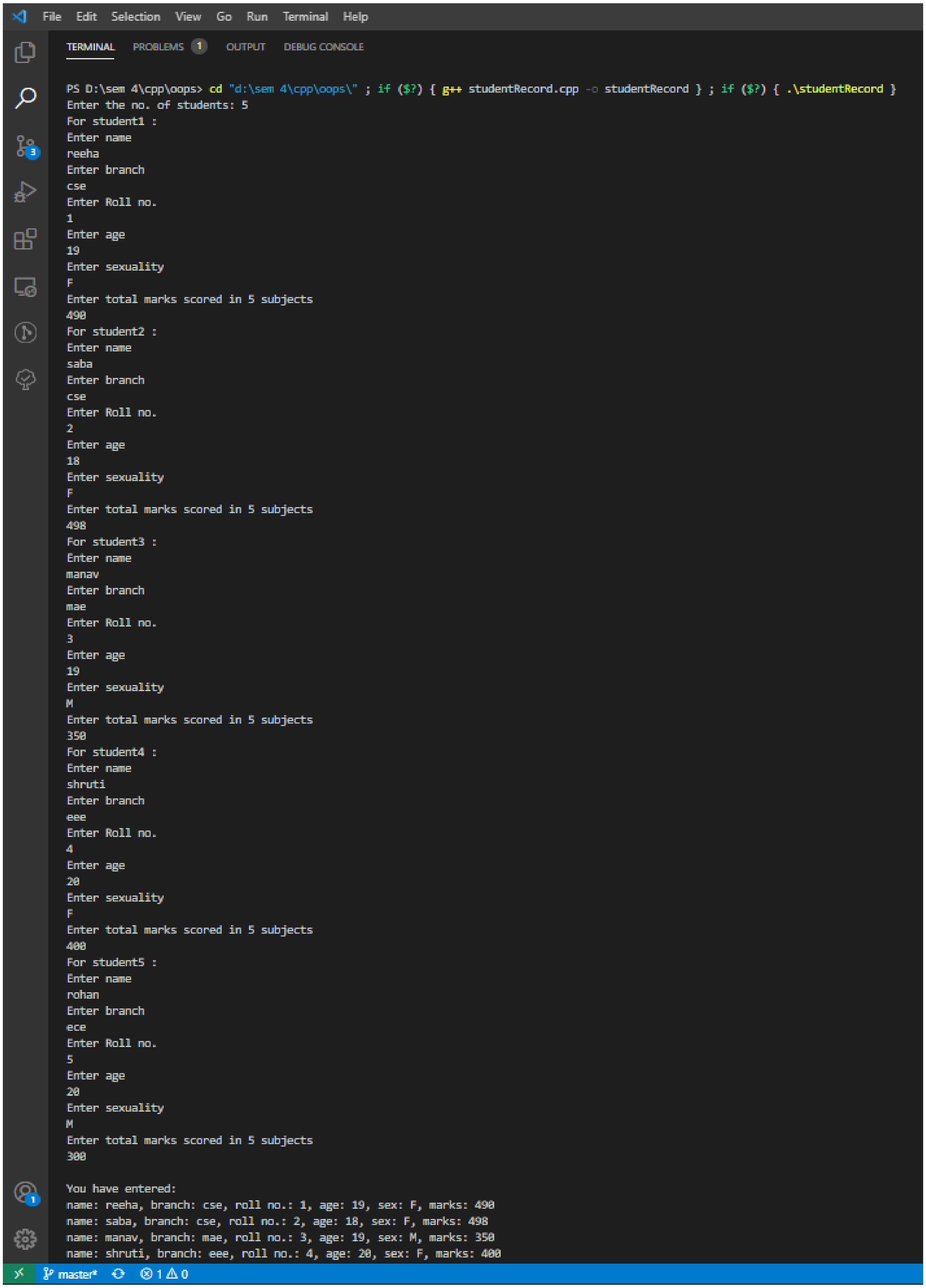
Aim:

Create a class student which have data members as name, branch, roll no., age, sex and marks in five subjects. Display the name of the student and his percentage who has more than 70%.

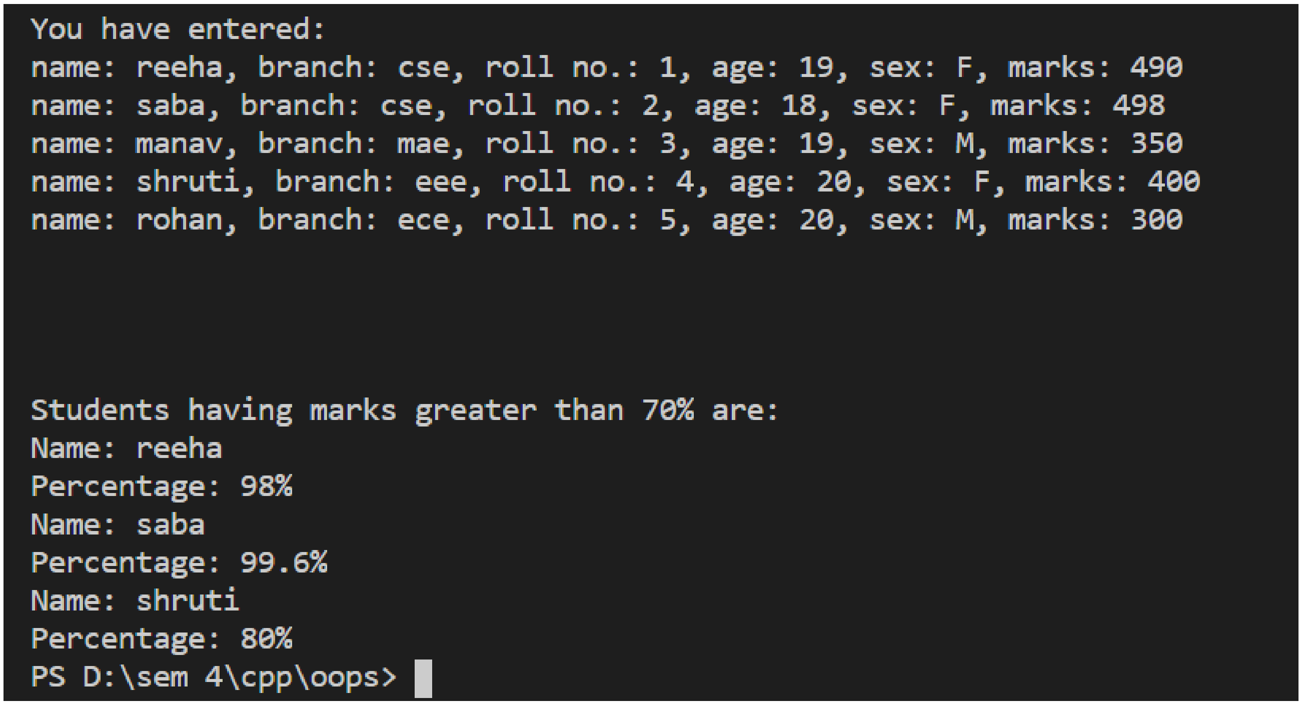
Source Code:

```
#include <iostream>
#include <string>

using namespace std;

class studentRecord{
    private:
        string name;
        string branch;
        int rollNo;
        int age;
        char sex[15];
        float marks;

    public:
        void getDetails();
        void check();
        void showDetails();
        void details();
};

void studentRecord::getDetails(){
    cout << "Enter name" << endl;
    cin >> name;
    cout << "Enter branch" << endl;
    cin >> branch;
    cout << "Enter Roll no. "<< endl;
    cin >> rollNo;
    cout << "Enter age" << endl;
    cin >> age;
    cout << "Enter sexuality" << endl;
    cin >> sex;
    cout << "Enter total marks scored in 5 subjects" << endl;
    cin >> marks;
}

void studentRecord::showDetails(){
    cout << "Name: " << name << endl;
    cout << "Percentage: " <<  (marks/5) << "%" << endl;
}

void studentRecord::check(){
    if (((marks/5)) > 70) {
        showDetails();
    }
}

void studentRecord::details(){
    cout << "name: " << name << ", branch: " << branch << ", roll no.: " << rollN
o << ", age: " << age << ", sex: " << sex << ", marks: " << marks << endl;
}

int main(int argc, char const *argv[]){
    int students;
    cout << "Enter the no. of students: ";
    cin >> students;
    studentRecord studentsArr[students];
    for (int i = 0; i < students; ++i) {
        cout << "For student" << i + 1 << " :" << endl;
        studentsArr[i].getDetails();
    }

    cout << "\nYou have entered:" << endl;
    for (int i = 0; i < students; i++) {
        studentsArr[i].details();
    }

    cout << "\n\n\n" << endl;

    cout << "Students having marks greater than 70% are: " << endl;
    for (int i = 0; i < students; i++) {
        studentsArr[i].check();
    }
}
```

Output:



```
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ studentRecord.cpp -o studentRecord } ; if ($?) { .\studentRecord }
Enter the no. of students: 5
For student1 :
Enter name
reeha
Enter branch
cse
Enter Roll no.
1
Enter age
19
Enter sexuality
F
Enter total marks scored in 5 subjects
490
For student2 :
Enter name
saba
Enter branch
cse
Enter Roll no.
2
Enter age
18
Enter sexuality
F
Enter total marks scored in 5 subjects
498
For student3 :
Enter name
manav
Enter branch
mae
Enter Roll no.
3
Enter age
19
Enter sexuality
M
Enter total marks scored in 5 subjects
350
For student4 :
Enter name
shruti
Enter branch
eee
Enter Roll no.
4
Enter age
20
Enter sexuality
F
Enter total marks scored in 5 subjects
400
For student5 :
Enter name
rohan
Enter branch
ece
Enter Roll no.
5
Enter age
20
Enter sexuality
M
Enter total marks scored in 5 subjects
300

You have entered:
name: reeha, branch: cse, roll no.: 1, age: 19, sex: F, marks: 490
name: saba, branch: cse, roll no.: 2, age: 18, sex: F, marks: 498
name: manav, branch: mae, roll no.: 3, age: 19, sex: M, marks: 350
name: shruti, branch: eee, roll no.: 4, age: 20, sex: F, marks: 400
```



```
You have entered:
name: reeha, branch: cse, roll no.: 1, age: 19, sex: F, marks: 490
name: saba, branch: cse, roll no.: 2, age: 18, sex: F, marks: 498
name: manav, branch: mae, roll no.: 3, age: 19, sex: M, marks: 350
name: shruti, branch: eee, roll no.: 4, age: 20, sex: F, marks: 400
name: rohan, branch: ece, roll no.: 5, age: 20, sex: M, marks: 300




Students having marks greater than 70% are:
Name: reeha
Percentage: 98%
Name: saba
Percentage: 99.6%
Name: shruti
Percentage: 80%
PS D:\sem 4\cpp\oops>
```

Viva Questions

1. What is an abstract class and when do you use it?

Ans.

A class is called an abstract class whose objects can never be created. Such a class exists as a parent for the derived classes. We can make a class abstract by placing a pure virtual function in the class.

2. What are destructors in C++?

Ans.

A constructor is automatically called when an object is first created. Similarly when an object is destroyed a function called destructor automatically gets called. A destructor has the same name as the constructor (which is the same as the class name) but is preceded by a tilde.